12

CLAIMS

1.   A computer system having a plurality of components that can be initialised,
5   wherein each component is configured to produce status data from which the level of
need for that component to be initialised can be inferred, the status data having a
predetermine level of need associable therewith, and wherein at least one component is
configured to: receive status data from other components; make a comparison using the
status data received from respective components; in dependence on the comparison,
10   select one or more components for initialisation; and, issue initialisation instructions to the
selected component(s).

2.   A computer system as claimed in claim 1, wherein the or each component
configured to make a comparison using status data is configured to use its own status
15   data in addition to the received status data when making the comparison.

3   A computer system as claimed in claim 1 or claim 2, wherein the components are
software components, and wherein the system includes at least one computer device on
which, in use, the software components are run.
20

4   A computer system as claimed in any of claims 1 to 3, wherein the status data is
in the form of an initialisation parameter.

5   A computer system as claimed in claim 4, wherein each component is configured
25   to execute an initialisation routine when the initialisation parameter for that component
reaches a respective threshold value, the initialisation routine including the step of
transmitting a request for an initialisation parameter to other components.

6   A computer system as claimed in claim 5, wherein the initialisation routine
30   includes the further steps of: receiving initialisation parameters from at least some of
those other components; comparing the received initialisation parameters with the
initialisation parameter for that component; and, in dependence on the comparison,
making a self-initialisation decision.

7   A computer system as claimed in any of claims 4 to 6, wherein each component includes a timer module for registering the elapsed time since the previous initialisation of that component, and wherein for each component, the initialisation parameter is determined at least in part in dependence on the elapsed time registered by the timer
5   module.

8   A computer system as claimed in any of claims 4 to 7, wherein each component is configured to produce an initialisation parameter that is at least in part dependent on whether the component is performing one of a number of predetermined tasks.
10

9   A computer system as claimed in any of claims 3 to 8, wherein the computer system includes a plurality of interconnected computer devices, each of which is housed in a respective housing, and wherein each device has, in use, a respective software component running thereon.
15

10.   A computer system as claimed in claim 9, wherein the software components each include a respective operating system module for operating the computer device on which the respective software component is running.

20   11.   A computer system as claimed in claim 9 or claim 10, wherein each component is configured to initiate a re-boot routine upon receipt of an initialisation instruction, the re-boot routine being configured to re-boot the computer device on which the software component is running.

25   12.   A computer system as claimed in claim 11, wherein the re-boot routine includes the step of determining if the computer device is performing a predetermined task or one of a number of predetermined tasks, and only to permit the re-booting of the computer device if the computer device is not performing such a task.

30   13.   A computer system as claimed in any of claims 3 to 8, wherein the components in use run on a common computer device, under the control of a common operating system.

14.   A computer system as claimed in any of claims 3 to 13, wherein each component, upon receipt of an initialisation instruction, is configured such that the
35   component is killed and subsequently restarted.

15.   A computer system as claimed in any preceding claim, including a computer device configured to allocate tasks to the components, such that a task allocated to one component is dependent on the task or tasks being performed by at least some of the
5   other components.

16.   A method of initialising the components of a computer system, the method including the steps of: receiving status data from a plurality of components, the received status data having associable therewith a predetermined level of need for a component to
10   be initialised; determining for each component, the need for that component to be initialised relative to the need for at least one other component to be initialised; and, initialising at least some of the components in dependence on their so determined relative need.

15   17.   A computer device having, in use, a software component running thereon, the software component being configured to: receive status data from a plurality of other components, the received status data having associable therewith a predetermined level of need for a component to be initialised; make a comparison using the status data received from the respective components; in dependence on the comparison, select one
20   or more components for initialisation; and, issue initialisation instructions to the selected component(s).

18.   A computer device as claimed in claim 17, wherein the software component running thereon is configured to produce status data from which the need for that
25   component to be initialised can be inferred.

19.   A computer device as claimed in claim 18, wherein the software component is configured to compare its self-produced status data with status data received from at least one other component when selecting one or more components for initialisation.
30
20.   A computer programme, the computer programme being configured, when loaded on a computer device, to: receive status data from a plurality of software components, the received status data having associable therewith a predetermined level of need for a component to be initialised; make a comparison using the received status
35   data from the respective components; in dependence on the comparison, select one or

more of the software components for initialisation; and, issue initialisation instructions to the selected software component(s).

21.   A computer programme as claimed in claim 20, wherein the computer programme is configured to issue initialisation instructions for re-booting the computer device or another computer device.

22.   A computer programme product stored on a computer-useable medium, the computer programme product having a program stored thereon as claimed in claim 20 or claim 21.

23.   A computer system having a plurality of components that can be initialised, wherein each component is configured to produce status data from which the level of need for that component to be initialised can be inferred, and wherein at least one component is configured to: receive status data from other components; make a comparison using the status data received from respective components; in dependence on the comparison, select one or more components for initialisation; and, issue initialisation instructions to the selected component(s).

24.   A method of initialising the components of a computer system, the method including the steps of: using status data from a plurality of components to determine for each component, the need for that component to be initialised relative to the need for at least one other component to be initialised; and, initialising at least some of the components in dependence on their so determined relative need.